032CM - 2025

PROGRAMMING FOR COMPUTATIONAL CHEMISTRY

Introduction

Gianluca Levi

gianluca.levi@units.it, giale@hi.is

Office: Building C11, 3rd floor, Room 329

Fall 2025

Register on **Moodle**

Join the **Teams page** of the course

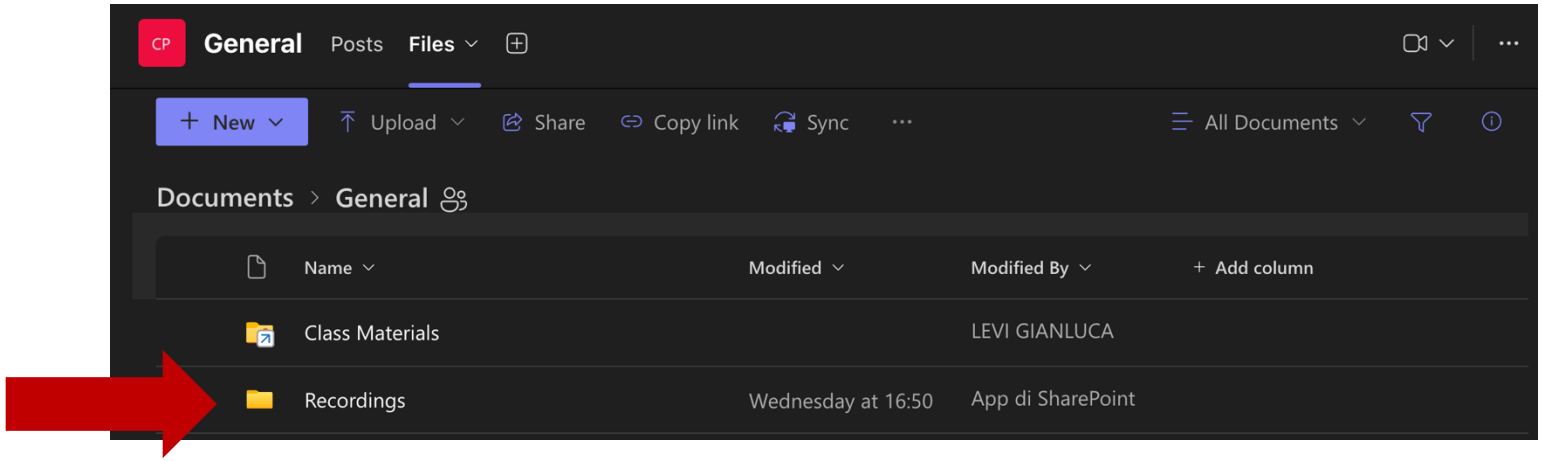
- Find the Teams code in the course catalogue

Anno Di Offerta: 2025/2026

CHIMICA (SM13A)

INSEGNAMENTO	CODICE TEAMS	PERIODO	DOCENTI
PROGRAMMING FOR COMPUTATIONAL CHEMISTRY (032CM - 2025 - [SM13A+4+ - ORD. 2025] ANALITICA E AMBIENTE - AC 1)	9cgie5v	S1	LEVI GIANLUCA
PROGRAMMING FOR COMPUTATIONAL CHEMISTRY (032CM - 2025 - [SM13A+5+ - ORD. 2025] NANOMATERIALI, ENERGIA E MODELLING - AC 1)	9cgie5v	S1	LEVI GIANLUCA

- Access **recordings of all lectures** in the Files section on Teams



Programme

Introduction to UNIX and Linux shell – Week 1

Fortran Basics – Weeks 1-2
Language structure, variables, arrays, subroutines, functions, simple I/O

Python Basics – Weeks 3-4
Data types, lists and dictionaries, loops, conditionals, functions and modules

Scientific Computing with Python – Weeks 5-6
Jupyter notebooks, NumPy, Matplotlib, debugging and profiling

Applications in Computational Chemistry with Python – Weeks 7–9

Python Project Structure – Weeks 10-11
Organizing projects, version control (Git), IDEs

Study material

Lecture slides

Online tutorials suggested by the teacher

Books

- Fortran for Scientists and Engineers, 4th edition, S. J. Chapman, McGraw-Hill Education
- Programming for Computations - Python, 2nd edition, S. Linge and H. P. Langtangen, Springer (Open Access)
- Python for Chemists, C. Hill, Cambridge University Press

Assignments (learn by doing!)

Problem sets provided by the teacher (**approx. one per week**)

- Start working on problems during hands-on sessions in class, then complete at home
- **Submit short report** to the teacher including input and output text
- You may collaborate on the homeworks, but submissions must be individual

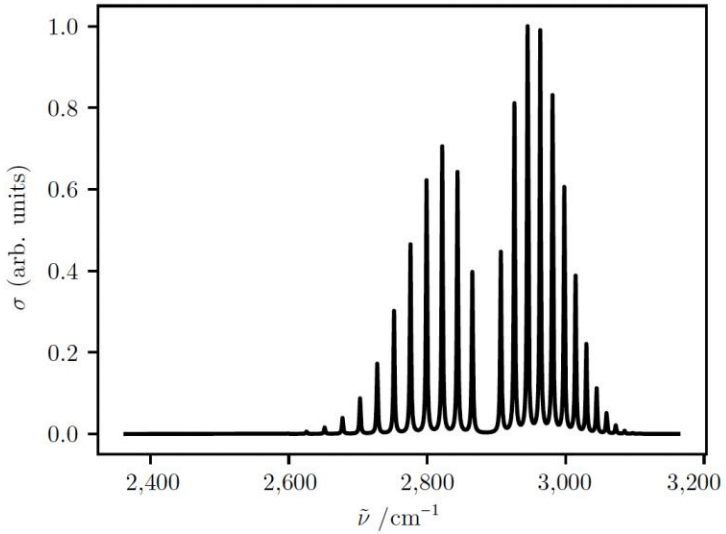
Assessment

	Grading
Weekly assignments (approx.) Computational exercises with Fortran and Python	***Hand in 70% of the assignments to qualify for the final exam***
Final exam (written) Write and execute Python code	50%
Final exam (oral) Discussion of the written exam and topics covered in the lectures	50%

Why learning programming for computational chemistry?

Why learning programming for computational chemistry?

Create computer-based tools to address tasks in chemistry (and beyond) in an **efficient**, **automated**, and **reproducible** manner



Simulated IR spectrum of $^{1}H^{35}Cl$ from Python for Chemists, C. Hill

Work with **large computational chemistry software packages** (e.g. atomic-scale simulation programs)

- **Understand** what happens under the hood
- **Adapt and modify** implementations for specific research questions

Why learning programming for computational chemistry in the AI era?

Why learning programming for computational chemistry in the AI era?

Use AI effectively: Write good prompts, evaluate and correct code generated by AI

Train the brain to **think algorithmically**

- Develop transferable **problem solving skills**
  - **Break down complex problems** into logical, algorithmic steps
- **Understand** computational methods and theory
  - **Gain deeper insights** into scientific problems